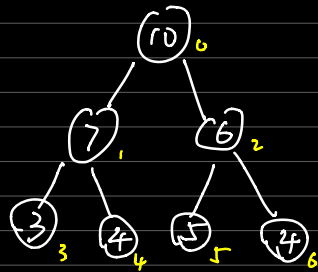


堆：一种特殊结构的二叉树，如下即为一堆，堆是一颗完全二叉树



其有两种构造方式。大根堆，小根堆。两种其实可以归结为一种

### 存储结构

完全二叉树一般使用数组来存，节省指针空间。关于完全二叉树的索引，其有如下

$$\text{parent}(i) = \text{floor}((i-1)/2)$$

$$\text{left}(i) = 2 \times i + 1$$

$$\text{right}(i) = 2 \times i + 2$$

$i$  为任意一个节点的 index，可以通过节点索引来找到父节点和子节点。

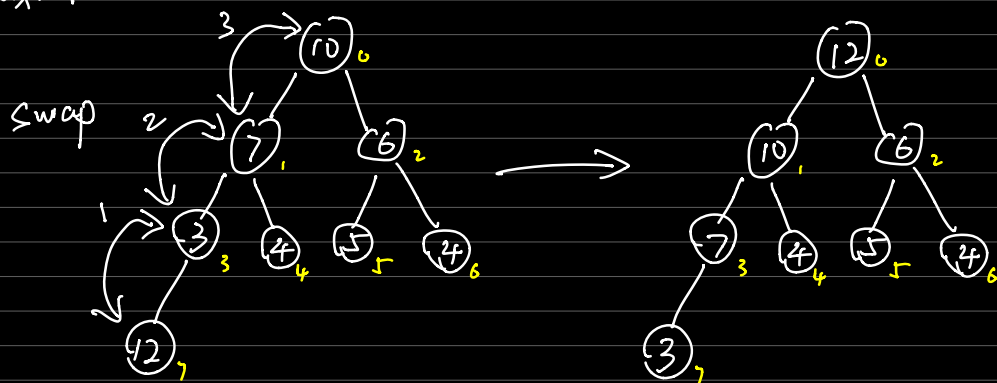
### 堆解决的问题

前  $k$  个元素动态维护的问题，其维护的复杂度为  $\log n$

### 堆的主要操作

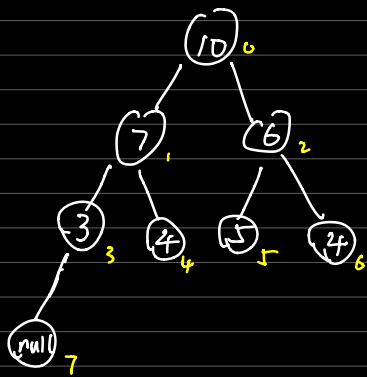
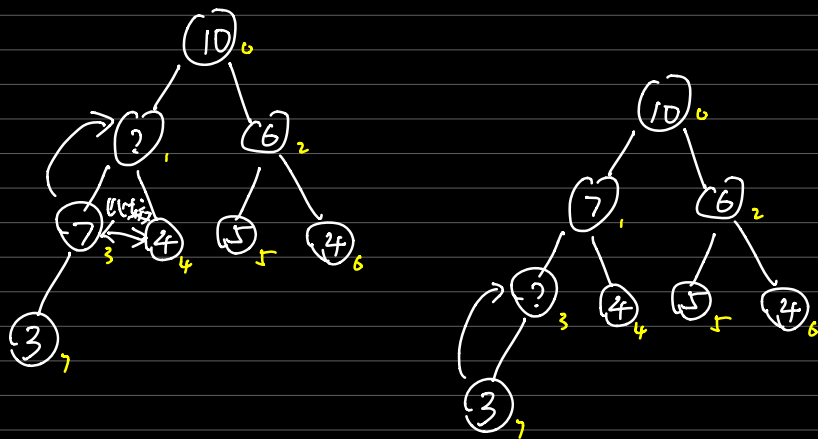
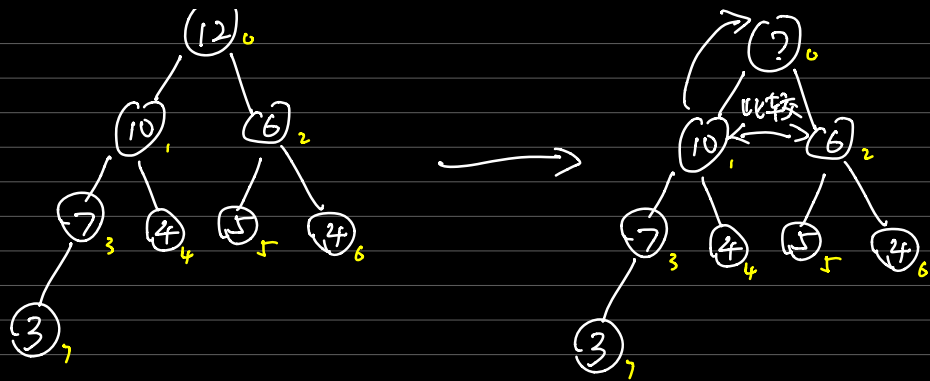
上浮和下沉，分别用于增加元素和删除元素。

#### 上浮操作



#### 下沉操作





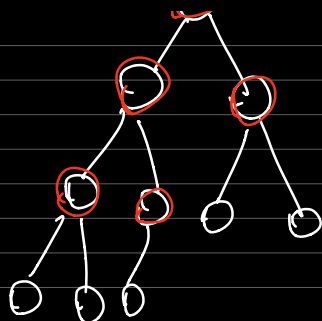
有了上面两个操作的支持,我们就可以完成增加和删除了.

增加就是和上浮一致,删除则是把删掉的顶端根换成 null 或者是数组的最后一个元素 然后一步步下沉即可.

无序建堆



对红色节点(从最后的非叶子节



点开始)进行下溢,就可以建堆  
3. 方法的复杂度是  $n$

堆排序,

有了前面的基础,堆排序是两部分

1. 建堆

2. 不断弹出最大的元素,

其复杂度为  $O(n + n \log n) = O(n \log n)$